

PATENT APPLICATION
IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of

Docket No: Q60558

Akio SUTO

Appln. No.: 09/819,612

Group Art Unit: 2161

Confirmation No.: 4173

Examiner: Etienne Pierre LEROUX

Filed: March 29, 2001

For: DISTRIBUTED DATA PROCESSING SYSTEM AND METHOD OF PROCESSING
DATA IN DISTRIBUTED DATA PROCESSING SYSTEM

REQUEST FOR REHEARING UNDER 37 CFR 41.52

MAIL STOP APPEAL BRIEF - PATENTS

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 41.52, Appellant respectfully submits this Request for Rehearing in response to the Decision on Appeal dated March 19, 2008. The points misapprehended and arguments overlooked are set forth below.

Claims 1-2, 7-9 and 16- 19 have been rejected under 35 U.S.C. § 102 as being anticipated by Mayhead (U.S.P. 6,367,029). Claims 3-4, 10-12 and 14 have been rejected under 35 U.S.C. § 103 as being unpatentable over Mayhead in view of Makinen (U.S.P. 5,758,067). Claims 5-6 have been rejected under 35 U.S.C. § 103 as being unpatentable over Mayhead and further in view of Nakamura (U.S.P. 5,347,463). Claims 13 and 15 have been rejected under 35 U.S.C. §

103 as being unpatentable over Mayhead in view of Nguyen (U.S.P. 6,202,070). Appellant respectfully requests reconsideration of the Decision on Appeal, affirming these rejections.

Appellant submits that the Decision on Appeal misapprehends points and overlooks arguments set forth in the Reply Brief at pages 3-4 and 5 (pages unnumbered in original Reply Brief filing). Applicant submits the points overlooked and arguments overlooked are as follows:

- 1) The Decision on Appeal does not adequately address the correct and most plausible reading of Mayhead such that the reference does **not** teach a replication trigger generator, based on updating of the database performed by clients connected to one of (multiple) servers as claimed. Appellant's Reply Brief as well as the Appeal Brief clearly raise a lack of express disclosure of the replication trigger generator based on update of the database by a client process and explain that the cited replication unit in Mayhead is a hardware or component manager.
- 2) The Decision on Appeal does not adequately address the correct and most plausible reading of Mayhead such that the reference does **not** teach updating information transfer unit for transferring updating information of said database to another one of the servers based on said replication trigger (which based on updating of said server database by process of the client). Appellant's Reply Brief as well as the Appeal Brief clearly raise a lack of express disclosure of the transfer of

updating information from one database to another server, based on updating of the database by processes performed by the client.

- 3) The Decision of Appeal does not adequately address the correct and most plausible reading of Mayhead such that the reference does **not** update the database (based on a process performed by a client) before generating the trigger signal.

As a matter of convenience, the first portion of the Reply Brief (pages 3-4) overlooked by the Decision is replicated below:

The replication manager also manages entry and exit of nodes into the system. Col. 2, lines 29-31. The nodes are computers or other hardware elements. Col. 2, lines 65-67. It is significant to note that the replication manager does not manage the database items, such as items stored in the file store. The management of that database is not performed through a replication trigger as claimed.

The Examiner appears to be confusing management of the replicable components (e.g. the file store, the checker and the logger) with replication management of objects within the database. In this regard, it is significant to note that claim 1 describes database updated by the clients which is described by recitation (1)¹ Claim 1 also describes updating of the database of another server based on the replication trigger which is described by

¹ (1) a replication trigger generator for generating a replication trigger based on the updating of said database by the distributed data processing process performed by said clients connected to one of the servers; (Emphasis added)

recitation (2).... (Emphasis added).² Therefore, the replication is in connection with update data provided by a client [as a result of a process performed thereby]. By contrast, the replication in Mayhead stems from the addition of hardware components (nodes) added to a system and not the data provided by a client.

To the extent any database management is suggested in Mayhead, it is performed via the checker and not the replication manager. See col. 2, lines 55-58 and col. 6, lines 44-50.

.....

The Examiner sets forth broad constructions of trigger and replication at pages 17-18 of the Examiner's Answer. Even assuming *arguendo* that these broad constructions are correct, nothing in Mayhead requires updating of the database to another server based on said replication trigger as claimed.

.....

It is also noted that while the Examiner has relied upon a logger as providing an archive data memory, the logger of Mayhead is not for recovery of database data as described in the last wherein clause of the claim. Rather, the logger only updates the status of nodes as they enter or exit the system. Col. 9, lines 62-64. Because updates of the database occur via the checker and not the logger, the logger cannot comprise the archive memory.

For all the above reasons, we would maintain that claim 1 is patentable and claim 7 is patentable for analogous reasons.

² an updating information transfer unit for transferring updating information of said database to another one of the servers based on said replication trigger.... (Emphasis original to Reply Brief)

First Set of Arguments and Facts Overlooked: The Decision on Appeal sets forth that the pivotal issue is whether the replication manager of Mayhead teaches the replication trigger generator as described by independent claim 1. The Decision relies on basically three citations to Mayhead at col. 2, lines 17-28; col. 3, lines 18-27 and col. 8, lines 23-44 as findings of fact to support the rejection. Appellant submits that these citations do not expressly teach all requirements of the claimed replication trigger generator. Moreover, the findings of fact completely ignore the Reply Brief citation to col. 2, lines 29-31 that indicate that replication management is a node management (hardware or software) and not database management.

Though the Decision relies on references to “coherency” and “consistency” to maintain the rejection, the Decision overlooks basic definitions of “coherency” in Mayhead. A basic flaw arises from the Decision’s overlooking the definition of “coherency” as related to the replication unit. Coherency, as defined in Mayhead relates to keeping track of which components are replicated, how many replicas currently exist and for teach replicated component whether replica is a “primary” component. Col. 7, lines 17-20. (Emphasis added). Components are described as hardware or node components. See col. 2, lines 29-34. This citation further describes the replication in connection with file store devices and logger devices, indicating that the replication manager tracks hardware (or software) components. Thus, the coherency maintained by the replication manager in Mayhead clearly relates to hardware (or software) management, as set forth by Appellant’s Reply Brief. By contrast, claim 1 describes “a replication trigger generator

for generating a trigger signal based on the updating of [a] database by the distributed data processing performed by said clients connected to one of the servers. “ Thus, the replication management for the entry and exit of service (e.g. hardware) nodes of Mayhead cannot correspond to a trigger based on updating of a database by distributed processing performed by clients. Thus, this is a first clear error in fact, and thus Appellant respectfully submits that the Decision be withdrawn.

Second Set of Arguments and Facts Overlooked: The Decision’s reliance on col. 8, lines 23-44 is similarly deficient. Col. 8 describes a client request for a specific service, with all service requests being first forwarded to a “primary” node. Col. 8, lines 28-29. Before the request is made, the primary will perform consistency and authorization checks. The replication manager (the component manager) as discussed above, will inform a logger component and a checker component of the request. If the logger component and the checker component must be replicated (in nodes or hardware), the (receiving) logger and checker are referred to as “primary”, as opposed to a “back up.” If necessary, the “back up” (hardware or software node) may need to be created as a result of the client request. Up to this point, neither the primary nor the client has performed any process that results in database update. The “consistency” check is only a hardware consistency and authorization check. The primary may be informed of the request, but the primary does not first perform the request. Rather, the “backups” are first requested to perform the request and inform the primary if the result of the process is successful. If all the backup server outcomes are successful, it is only then that the primary (the node

receiving the request) will perform the request locally itself and update and store the result and inform the client of the result.

In relying on Col. 8, the Decision continues to confuse the role of a replication manager for any management of a data update due to a process performed by the client. The above discussion clearly indicates that the process is performed at process server elements, not client elements as claimed. This point was raised in the Reply Brief, but has been overlooked by the Decision's over-reliance on the purported replication manager. In Mayhead, only when the backup processors successfully performs the operations and returns a successful "indicator" to the primary is the database updated, and then a result is sent to the client. By contrast, as emphasized in the Reply Brief, claim 1 describes a database which is updated by the distributed process performed by the client. It is clear in Mayhead that the client receives a processing result, but itself does not perform a process to update the database.

From the above discussion, as raised in the Reply Brief, the Decision's own citation at col. 8 cannot support the rejection. In Mayhead, update information is not even transferred from server to server based on the replication trigger, but processes are performed locally and repetitively. Col. 8, lines 39-44. It is quite clear that such processes are not performed at the primary if the backups do not each return a "successful" indicator. Thus, there is a clear cut off between any purported replication trigger and an update of the database in another server. Therefore, there are alternative ways to update multiple databases that do not necessarily include, suggest or teach that which is recited by claim 1.

As a related matter, to the extent that an update is made of a (primary) database that receives the client request, the database update is made only after all of the following occurs: the backups perform a process, the backups send a successful indicator, the primary locally performs the process previously performed at the backup, and the date is made at the primary. Updating information is not sent to the primary database. Rather, only “successful indicators” are sent, and with the successful process performed at the backup, the primary performs its own process before updating its own database. The primary then returns the result to the client. The inter-relatedness of the database update upon client process, the generation of the trigger signal as a result of the update of database of the client process, and the transfer of information to update the other server based on the trigger, described by claim 1 clearly does not exist in the isolated citations relied upon in the Decision.

In the haste to conclude that multiple database systems need to maintain agreement in the contents of their databases, the Decision fails to specifically articulate a rejection that takes into account how the inter-relatedness of Appellant’s claim elements is taught in the art, as explained and parsed in the Reply Brief. Col. 12, lines 39-68 of Mayhead clearly delineate how Mayhead accommodates a client write request, i.e. a database change. It is clear at step 1 that a primary receives a request. However, the request is not updated at the primary (database until step 8). Rather the primary sends the request to the backups after requisite authorization (steps 2-3), the backup overwrites its own data (step 4). The backup sends an acknowledgement to the primary

that the update was successful (step 6), and only then is the primary node allowed to perform a process and store the new data (step 8).

Therefore, claim 1 is patentable, and by analogy, claim 7 is also patentable.

The discussion at page 5 of the Reply Brief outlines the third point overlooked in this Appeal. The relevant discussion is set forth below:

With further regard to the arguments for claim 19, claim 19 describes that update of the database occurs prior to the generation of the trigger. The Examiner cites col. 8, lines 23-45 of Mayhead to teach this feature. However, col. 8, lines 28-30 specifically states that the primary prepares for executing of a request. At that point, no update is performed. The replication manager (the purported trigger generator) then informs the logger and checker of a request. Col. 8, lines 31-35. It is only after the operations of the trigger manager (via the logger and checker) that the primary performs the update. Col. 8, lines 40-45. The Examiner's reliance on col. 8 actually supports reversal of the rejection of claim 19.

Third Set of Arguments and Facts Overlooked: With further regard to claim 19, this claim describes replication of a trigger occurs after update of the database, (where the update of the database is based on a process performed by the client). As discussed above, in Mayhead, the updating of the database (which receives a client request) is not updated until the last step of an update process. Thus, in Mayhead, any purported trigger will occur prior to the update. Therefore, claim 19 is patentable for this additional reason.

REQUEST FOR REHEARING UNDER 37 CFR 41.52
Application No.: 09/819,612

Attorney Docket No.: Q60558

The remaining claims are patentable based on their dependencies on claims 1 and 7.

In view of the above, Appellant respectfully submits that the discussions set forth in the Decision do not demonstrate an appreciation for the proper reading and operation of the Mayhead reference, and why the Mayhead reference does not expressly or inherently teach all aspects of the claims. Accordingly, withdrawal of the Decision is respectfully requested, such that the pending claims may be passed to issue.


Respectfully submitted,

SUGHRUE MION, PLLC
Telephone: (202) 293-7060
Facsimile: (202) 293-7860

WASHINGTON OFFICE

23373

CUSTOMER NUMBER


Susan Perng Pan
Registration No. 41239

Date: May 19, 2008